Instructional Videos and Interactive Notebooks for Learning of Coding Concepts in Chemical Engineering Analysis

Matthew Wilhelm, PhD Candidate
Chenyu Wang, PhD Candidate
Matthew Stuber, Assistant Professor
Challenges: Content

- Show students how to formulate and solve engineering problems
- Introduce concepts of algorithms
- Introduce numerical methods
- Teaching a “new” programming language

![General CHEG Curriculum](image)
Challenges: Content

- Show students how to formulate and solve engineering problems
- Introduce concepts of algorithms
- Introduce numerical methods
- Teaching a “new” programming language
  - *Intro to Computing* language determined by different department
  - 2 – 4 year gap after *Intro to Computing*
  - Interceding courses have minimal programming content
Approach

**New Interactive Modalities**
- Student generated videos
- Interactive notebooks & assignments

**Explicit Skills Focus**
- Teach explicit coding skills and content
- Implicit comprehension monitoring

**“Real World” Content**
- Complex examples
- Literature and industry content

An integrated approach to improvement
Content Gaps

**Computational Thinking**

- Problem formulation
- Recursion
- Problem Decomposition
- Abstraction
- Systematic testing
  - Debugging
  - Modular Programming


Notebook Design

- Designed notebooks for both Matlab (Live Editor) and Julia (Jupyter) initially

- A single document consisting of cells of either:
  - Rich text
  - Code

- Cells can be run individually or as a whole document
Tutorial Design #1

- **Statement of unit level objectives**

- **Brief Review of Theory**
  - Links to website detailing industry applications and relevant academic research.
  - Comments on extension material.

- **1 – 2 Problems**
  - Background statements to frame a problem in an industrial context.
  - Interactive content.
  - Commentary on coding fundamental interlaced throughout the example.
  - Prompts for reflection interspersed.

- **End with reflective questions.**
Jupyter Notebooks

- Supports 100+ languages: Matlab, Julia, Python, etc.

- Containerization support for open-source languages: Both free and commercial support (Docker, MyBinder, etc).

- More features via extensions
  - Hyperlinking to videos
  - Richer text formatting options

- Simple installation for open-source languages

- Difficult installation with Matlab
  - Install python -> Add/Update Packages Manually -> Change Environmental Variables
  - OS specific differences

- Minimal formatting restrictions.
Matlab Live Editor

- Available with Matlab Installation.

- Limited support for Rich text:
  - Latex, Images, Figures
  - Not as readily extendible, no HTML support, embedding videos is tricky.

- Live functions need to be placed at the bottom of the page.
  - Modular programming in this environment must break the flow.
  - Can’t have students **effectively** introduce functions in intermediate cells.
Interactive notebooks have been distributed this Fall semester as supplemental material.

Usage information is encouraging (15% of access to content, nearly all students, multiple uses).
We take a preliminary look at correlation between interactive notebook usage and examination scores for the first two exams. Equal frequency binning applied based on interaction number. We see that low usage of the notebooks correspond to with lowest exam performance in each case.

### Interactions vs. Exam 1

<table>
<thead>
<tr>
<th>Interaction</th>
<th>Average Score (n)</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOW (&lt;= 5)</td>
<td>65.8% (24)</td>
</tr>
<tr>
<td>MED (5 to 9)</td>
<td>71.9% (25)</td>
</tr>
<tr>
<td>HIGH (&gt; 9)</td>
<td>71.5% (24)</td>
</tr>
</tbody>
</table>

### Interactions vs. Exam 2

<table>
<thead>
<tr>
<th>Interaction</th>
<th>Average Score (n)</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOW (&lt;= 2)</td>
<td>61.7% (28)</td>
</tr>
<tr>
<td>MED (3 or 4)</td>
<td>77.3% (24)</td>
</tr>
<tr>
<td>HIGH (&gt; 4)</td>
<td>70.7% (21)</td>
</tr>
</tbody>
</table>
Undergraduate teaching assistants generated short topic primer videos (1 – 5 minutes). Scripted developed by student and workshopped prior to recording content.

Contains an example in which a student solves a sample problem while providing exposition.

Content driven slides are interspersed. This helps highlight the role of problem formulation and abstraction-based thinking while modelling an approach rooted in computational thinking.
## Student Videos – Preliminary Data

### Final Grades by Interactions w/ Video Content

<table>
<thead>
<tr>
<th>Number of Interactions</th>
<th>2017</th>
<th>2018</th>
<th>2019</th>
<th>2020</th>
</tr>
</thead>
<tbody>
<tr>
<td>No Interaction</td>
<td>62.5%</td>
<td>66.4%</td>
<td>64.0%</td>
<td>N/A*</td>
</tr>
<tr>
<td>One Video</td>
<td>N/A</td>
<td>N/A</td>
<td>70.4%</td>
<td>N/A*</td>
</tr>
<tr>
<td>Both Videos</td>
<td>N/A</td>
<td>N/A</td>
<td>74.8%</td>
<td>N/A*</td>
</tr>
<tr>
<td>All</td>
<td>N/A</td>
<td>N/A</td>
<td>72.1%</td>
<td>N/A*</td>
</tr>
</tbody>
</table>

* Course re-organized to a learning module format due to anticipated online delivery. Interaction with both videos increased to 96% (4.4 views on average).

### Interactions vs. Exam

<table>
<thead>
<tr>
<th>Interaction</th>
<th>Average Score (n)</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOW (&lt;= 6)</td>
<td>61.7% (25)</td>
</tr>
<tr>
<td>MED (7 to 10)</td>
<td>77.3% (24)</td>
</tr>
<tr>
<td>HIGH (&gt; 10)</td>
<td>70.7% (24)</td>
</tr>
</tbody>
</table>

Equal frequency binning applied based on interactions with Videos.
# Next Steps

<table>
<thead>
<tr>
<th><strong>Initial Design</strong></th>
<th>Instructional videos have been recorded and preliminary version of the interactive notebooks were constructed.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Preliminary Trial</strong></td>
<td>Instructional videos and literate programming notebooks have been included as supplemental material within existing learning modules for this Fall semester.</td>
</tr>
<tr>
<td><strong>Feedback</strong></td>
<td>Usage statistics are currently being monitored and a preliminary survey will collect student feedback pertaining to the interactive notebooks.</td>
</tr>
<tr>
<td><strong>Finalize</strong></td>
<td>Interactive notebooks will be updated based on student feedback</td>
</tr>
<tr>
<td><strong>Distribute</strong></td>
<td>Interactive notebooks of both the Matlab Live Editor® and Jupyter notebooks formats will be distributed through the CACHE organization.</td>
</tr>
<tr>
<td><strong>Rollout</strong></td>
<td>Interactive notebooks will become required material and completion thereof will be linked to assessment.</td>
</tr>
</tbody>
</table>
Acknowledgements

Undergraduate Teaching Assistants - Notebook Editing
- Megan Walsh
- Talha Bhatti

Undergraduate Research Assistant – Video Content
- Dimitri Alston

Funding
Provided by grants from the Center for Excellence in Teaching and Learning at the University of Connecticut (UCONN CETL) and Computer Aids for Chemical Engineering (CACHE)